

Polecenie: Napisać program, który czyta ciąg liczb ze standardowego wejścia aż do wystąpienia

znaku końca pliku (kombinacja klawiszy Ctrl-D na klawiaturze), a następnie oblicza średnią arytmetyczną i odchylenie standardowe, po czym wyprowadza wyniki na standardowe wyjście. W pierwszej wersji programu, dane mogą być przechowywane w jednowymiarowej tablicy liczb zmiennoprzecinkowych o ustalonym rozmiarze maksymalnym (np. $N_{max} = 1000$) zadeklarowanym jako stała całkowita. Proszę poeksperymentować z typami wprowadzanych danych (float i double) oraz formatami wprowadzanych i wyprowadzanych danych. Gotowy program proszę wykorzystać do czytania danych z pliku, z wykorzystaniem mechanizmu potoków unixowych (np. `./srednia.out < moje_dane.dat`).

Na początku programu zdefiniowałem standardową bibliotekę `<stdio.h>`, bibliotekę do funkcji matematycznych `<math.h>` oraz bibliotekę `<stdlib.h>`. Dalej zdefiniowałem funkcję `int main` w której będę wykonywał zadanie.

Następnie zadeklarowałem zmienne do liczb przecinkowych takie jak:

```
double numbers[1000], n, suma = 0.0f, srednia, odchy = 0.0f, odchylenie;
```

oraz zmienne do liczb całkowitych:

```
int i = 0, p;
```

`number[1000]` tablica na 1000 liczb, `n`-zmienna do przechowywania liczby wczytanej z pliku, `suma`-suma liczb, `srednia`-średnia liczb, `odchy`-zmienna pomocnicza do liczenia odchylenia, `odchylenie`-odchylenie standardowe.

Następnie wczytuje dane z pliku przy użyciu mechanizmów potoków unixowych,

```
FILE *file=stdin;
```

oraz definiuję pętlę `while` która zakończy się jeśli napotka na znak końca pliku EOF

```
while (fscanf(file, "%lf", &n) != EOF) /*jesli znak bedzie == EOF koniec petli*/
```

oraz wpisuje liczby do tablicy `number[]` oraz liczę bezpośrednio sumę liczb `suma=suma+n;`

Następnie zamykam plik z którego pobierałem dane.

W dalszej części programu obliczam średnią arytmetyczną przy pomocy wzoru:

```
srednia = (suma / i);
```

gdzie `suma` to suma wszystkich liczb a `i` to ilość dodanych liczb oraz wypisuje przy pomocy standardowego wyjścia tą średnią.

Następnie obliczam odchylenie standardowe:

Zdefiniowałem pętlę for która kończy się jeśli zmienna pętla będzie równa ilości wszystkich liczb. W samej pętli dokonuje obliczeń pomocniczych takich jak:

```
odchy = odchy + pow(numbers[p] - srednia, 2); /*obliczenie pomocnicze*/
```

czyli do zmiennej pomocniczej odchy dodaję ją samą oraz potęgę stopnia 2 z danej liczby wczytanej z tablicy-średnią arytmetyczną wszystkich liczb.

Następnie obliczam odchylenie standardowe używając wzoru:

```
odchylenie = sqrt(odchy / i);
```

czyli obliczam pierwiastek kwadratowy ze zmiennej pomocniczej odchy którą dzielę przez ilość wszystkich wczytanych liczb oraz wypisuję te dane przy pomocy standardowego wyjścia.

URUCHOMIENIE PROGRAMU:

```
bartosz@Baxing-VB:~/Desktop/C/Zestaw1$ gcc zad1.c -o zad1 -lm -std=c89
```

naależy dopisać -lm aby móc używać funkcji z biblioteki matematycznej takie jak pow() oraz sqrt()

PRZYKŁADY DZIAŁANIA PROGRAMU:

```
bartosz@Baxing-VB:~/Desktop/C/Zestaw1$ ./zad1<zad1n1.txt  
  
Srednia 4 liczb wynosi: 2.750000  
  
Odchylenie standardowe z 4 liczb wynosi: 1.089725
```

Dla liczb:

1
3
3
4

średnia arytmetyczna to:

$1+3+3+4=11$

$11/4=2.75$

odchylenie standardowe to:

pierwiastek z $((1-2.75)^2+(3-2.75)^2+...)$ podzielone przez ilość liczb czyli 4)
a więc wynikiem będzie 1.089725

DZIAŁANIE DLA LICZB NIE TYLKO CAŁKOWITYCH:

```
bartosz@Baxing-VB:~/Desktop/C/Zestaw1$ ./zad1<zad1n2.txt  
  
Srednia 5 liczb wynosi: 22.499980  
  
Odchylenie standardowe z 5 liczb wynosi: 28.738338
```

Liczby:

1.5

1.1

10.8

78.1

20.9999

DZIAŁANIE DLA LICZB RZECZYWISTYCH:

```
bartosz@Baxing-VB:~/Desktop/C/Zestaw1$ ./zad1<zad1n3.txt
```

```
Srednia 4 liczb wynosi: 1.250000
```

```
Odchylenie standardowe z 4 liczb wynosi: 12.417226
```

10

15

-17

-3

(PRZYKŁADOWE PLIKI Z LICZBAMI PRZESŁAŁEM DODATKOWO DO ZADANIA SĄ TO PLIKI: zad1n1.txt , zad1n2.txt oraz zad1n3.txt